
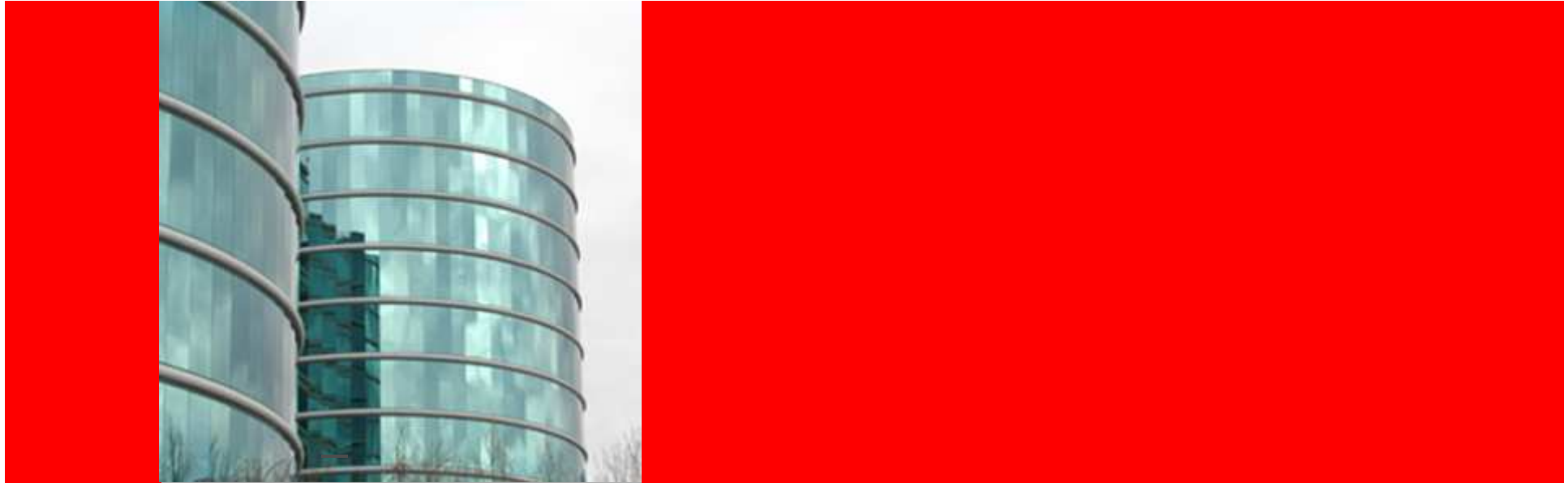


ORACLE®



The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.



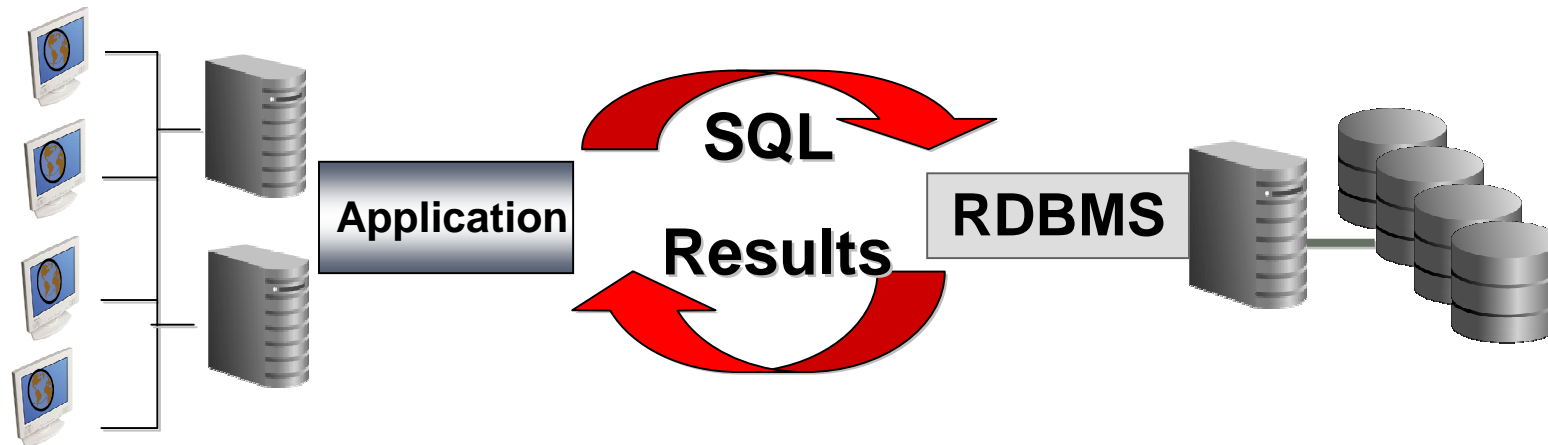
ORACLE[®]



Introduction to Oracle In-Memory Database Cache

Simon Law
Product Manager

When You Think “Database Application”



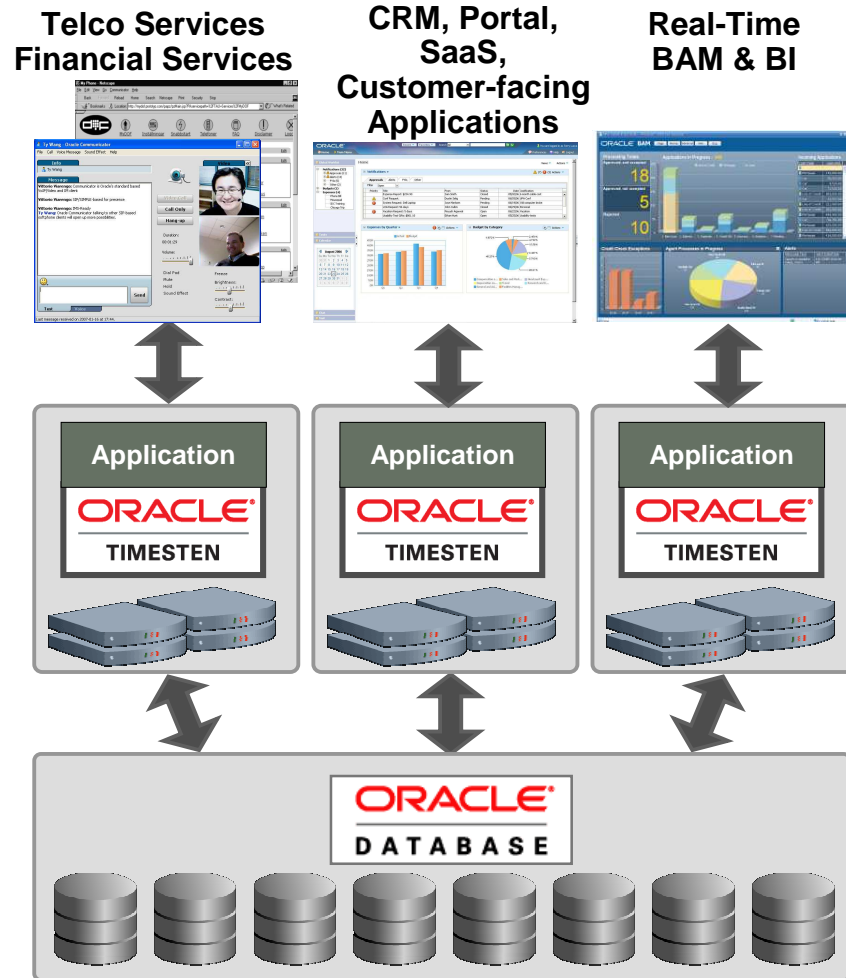
RDBMS and network connectivity

**NOT fast enough for some
Response time critical applications**

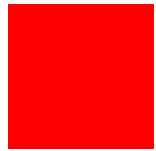
Typical solutions: *Build a home-grown, application-specific, in-memory buffer ‘cache’.....*

Oracle TimesTen In-Memory Database

Memory-Optimized RDBMS for Real-Time Applications

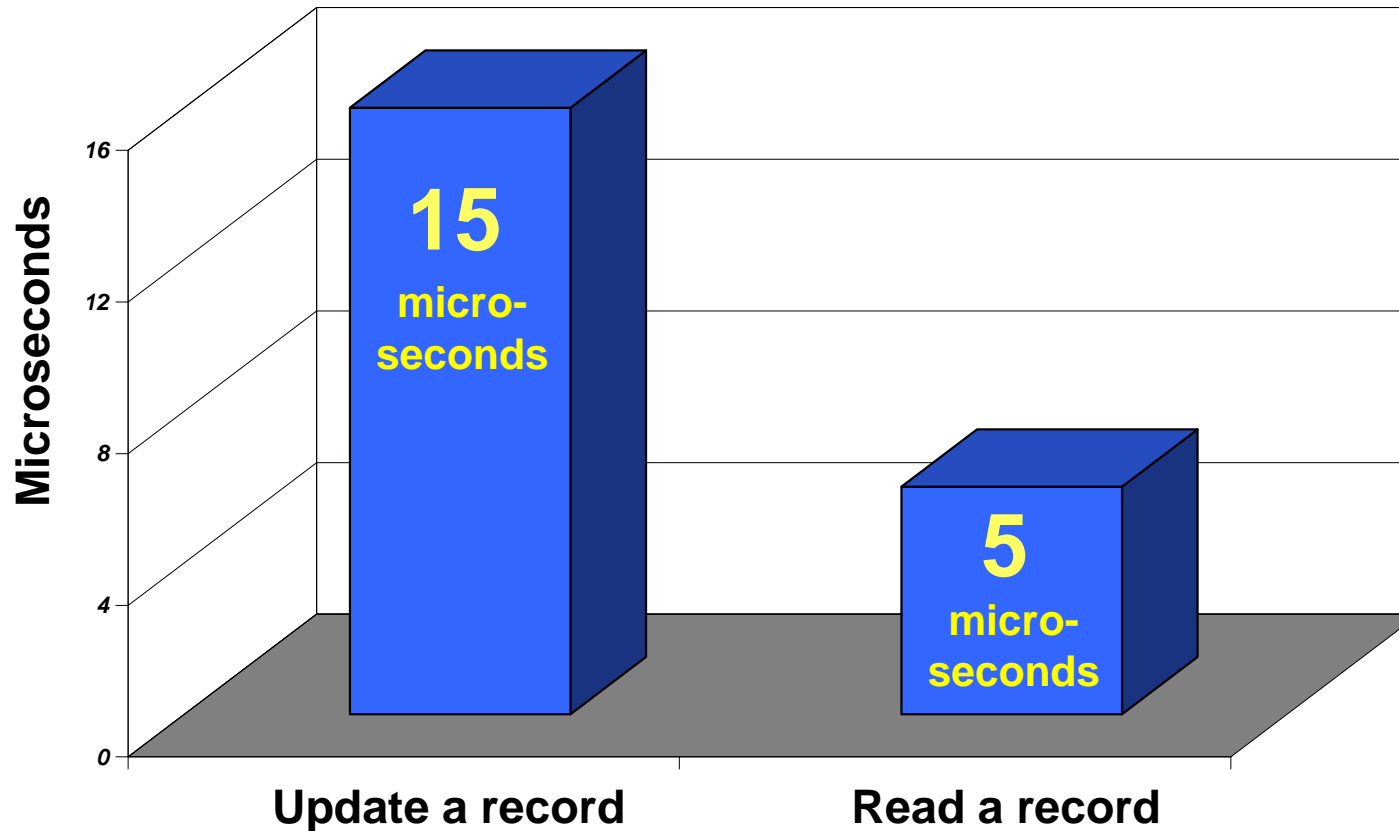


- Delivers *real-time responsiveness* and very high throughput
- Deployed in the application tier, as
 - Standalone in-memory database
- OR
- Read/Write In-memory database cache for Oracle Database

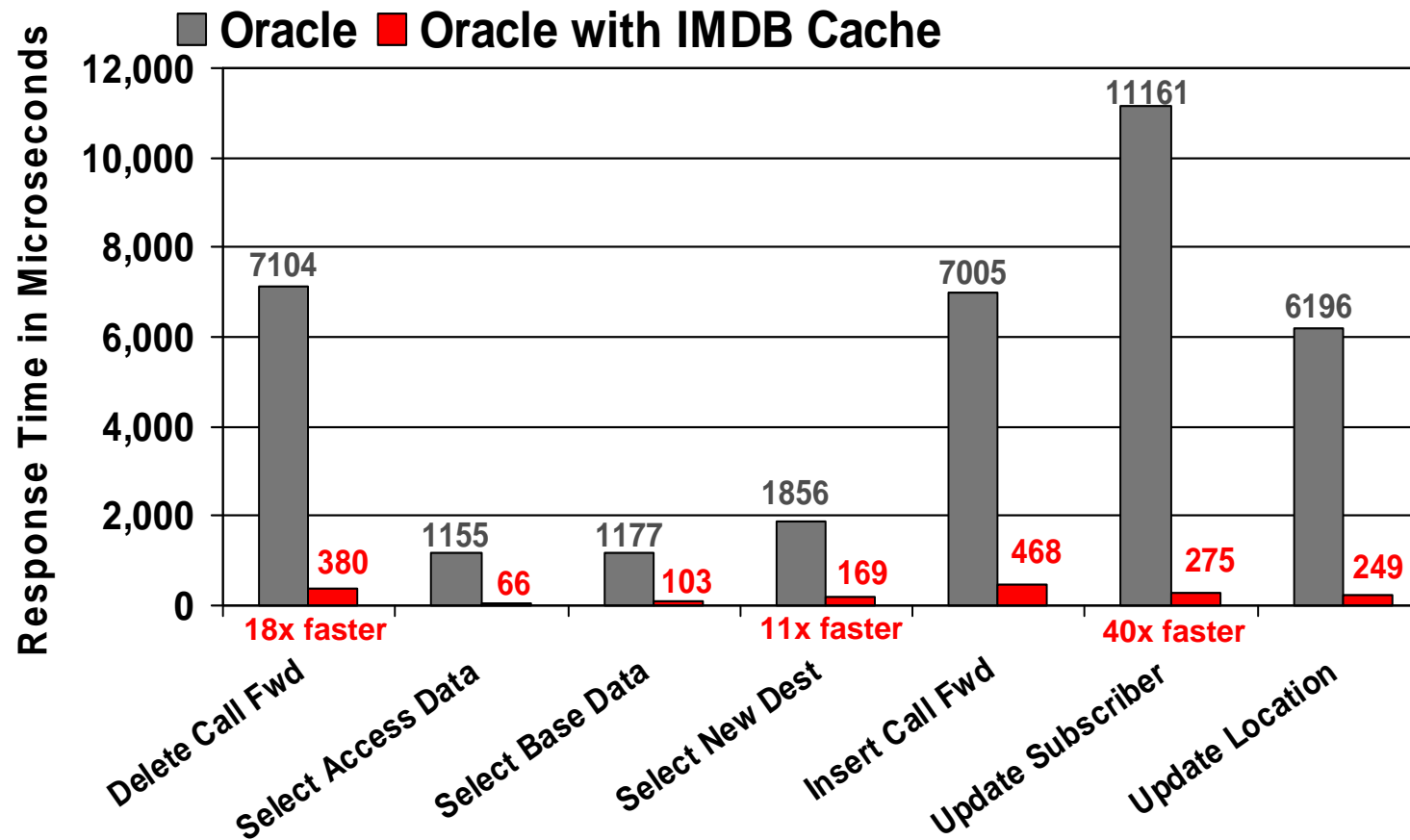


Lightning Fast Response

Average Response Time
TimesTen In-Memory Database



HLR Application Demo



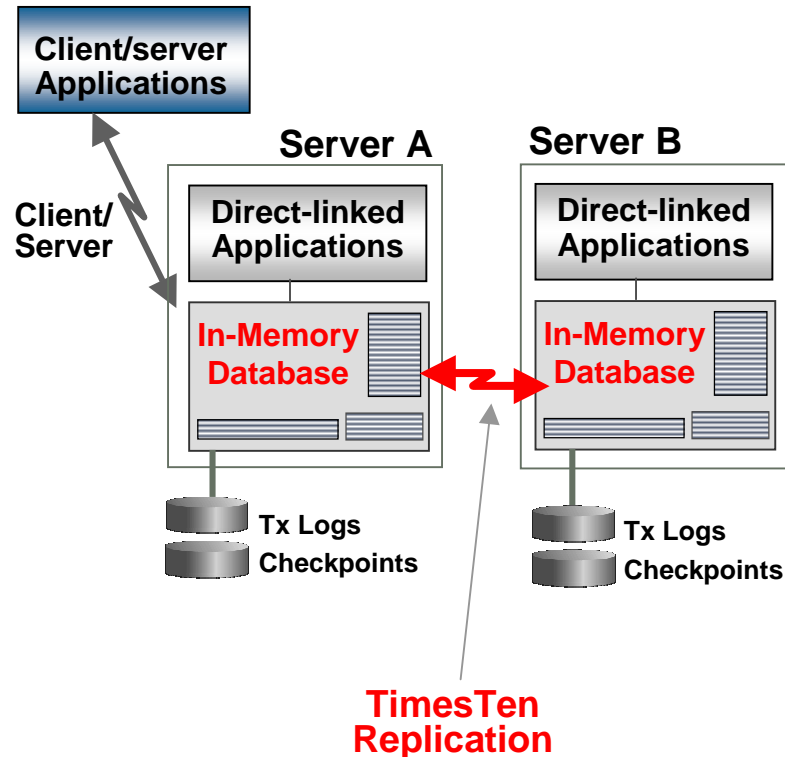
2-CPU, 2.4 GHz Intel P4/Xeon with 6GB of RAM, SLES9

Why Is TimesTen So Fast?



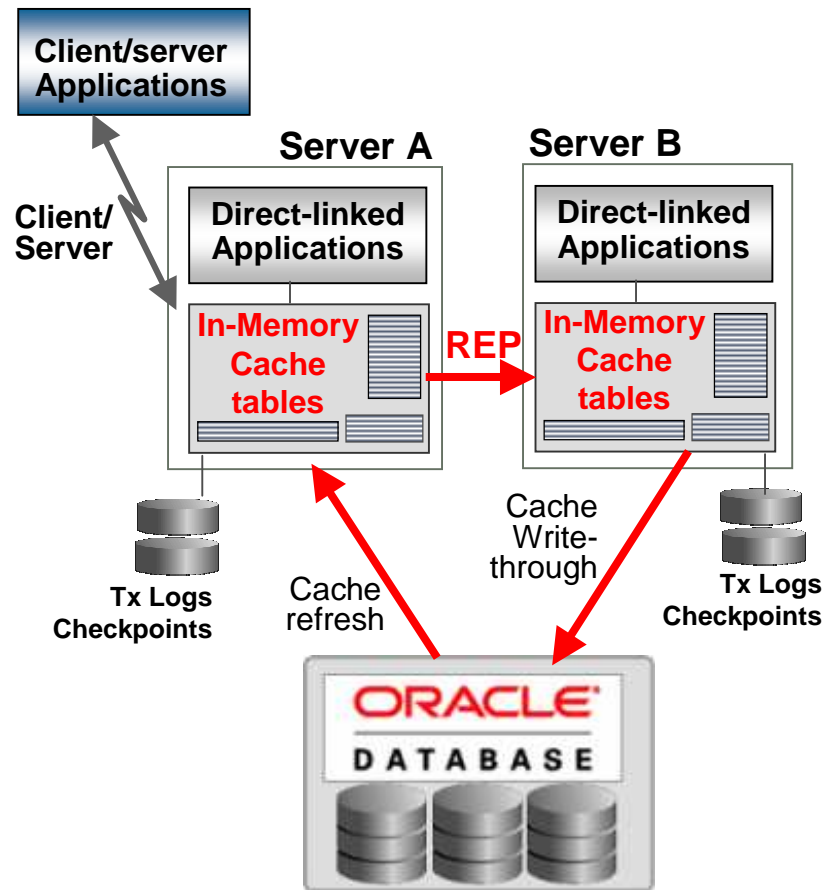
- In-Memory Database
 - Entire database is always in memory
 - Designed and optimized for memory layout
 - No buffer cache management overhead
 - Shorter code path = faster performance
 - Fewer CPU instructions = fewer processors required
- Application can embed in-memory database into its process address space
 - Eliminate network and inter-process communication overhead
 - Very low response time (like calling a procedure)

Oracle TimesTen In-Memory Database



- In-memory RDBMS
 - Entire database in memory
 - Standard ODBC/JDBC, SQL 92
 - Compatible with Oracle Database
 - Accessed via direct-linked or via client/server connections
- Exceptional performance
 - Very low response time → high throughput
- Persistence and recoverability
 - Database persists to disk
 - Transactions with ACID properties
- High Availability via transactional data replication

Oracle In-Memory Database Cache



- Improves application response time
- Cache tables, subset of rows and columns
 - Unlimited reads and writes
- Automatic data synchronization
 - Updates in cache are propagated to Oracle database
 - Updates in Oracle database are refreshed to in-memory cache tables
- Applications access in-memory cache tables like regular database tables using SQL
 - Joins/search, insert/update/delete



Types of In-Memory Caches

- Read-only Cache
 - Oracle Database updates automatically refreshed to in-memory cache tables
 - Updatable Cache
 - Transactions committed in the cache and write-through to Oracle Database
 - Asynchronous and synchronous write-through options
-
-
- Static Cache
 - Entire content of cache is pre-loaded prior to doing any work
 - Dynamic Cache
 - Data loaded transparently from Oracle if not found in the cache
 - Automatic data aging : LRU & Timestamp based aging
-
-

Multiple Read-Only Caches

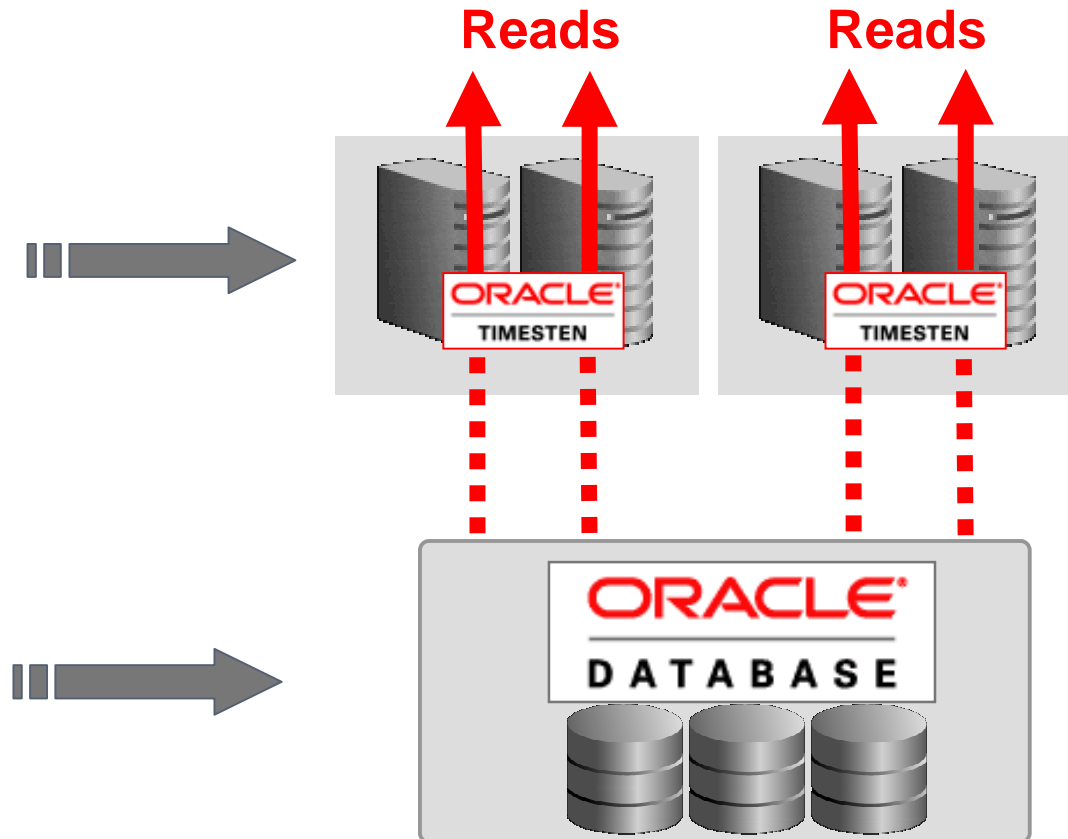
Examples

In-Memory Cache:

- ❖ User profile
- ❖ Ticket symbols
- ❖ Service policies
- ❖ Flight schedule

Oracle Database:

- ❖ Flight reservation
- ❖ Reference data
- ❖ Historical data data



Reference-data is heavily accessed by applications, “read-only”, and an ideal candidate for real-time caching

Multiple Updatable Caches

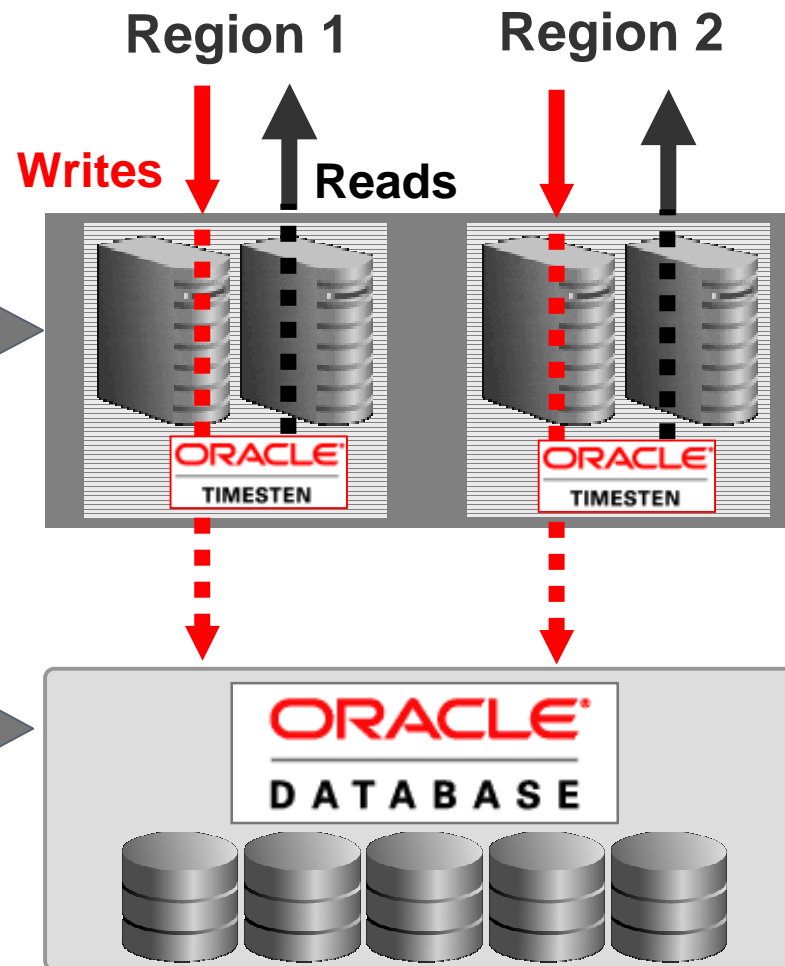
Examples

In-Memory Cache:

- ❖ Pre-paid billing data
- ❖ Portfolio/positions
- ❖ Account balance
- ❖ Flight reservation

Oracle Database:

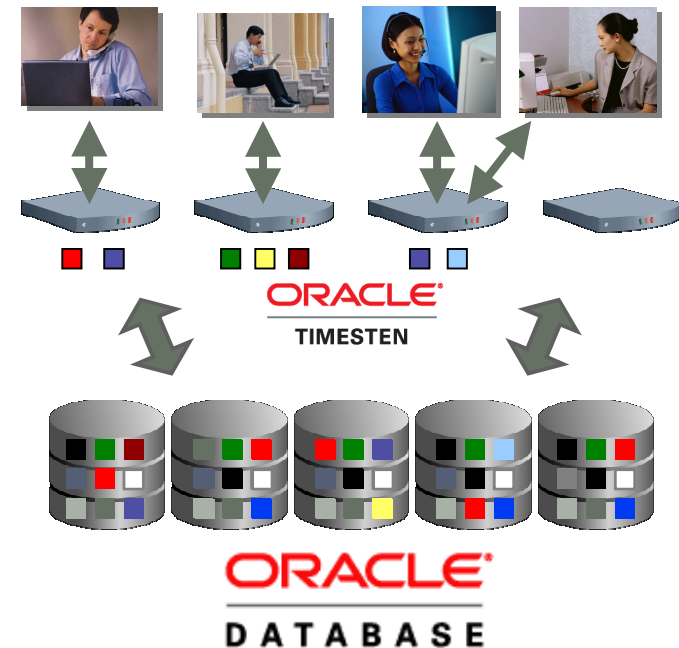
- ❖ DB of record
- ❖ Provisioning
- ❖ Order history
- ❖ Message storage



Dynamic Database Caching

Call Center Application Example

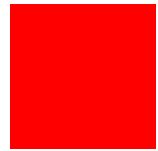
- Transparent loading of customer data from Oracle Database
 - Load customer data dynamically at the time of the call
 - Improve database responsiveness for subsequent operations
- Automatic data aging
 - Remove old or least-recently-used data to make room for new callers



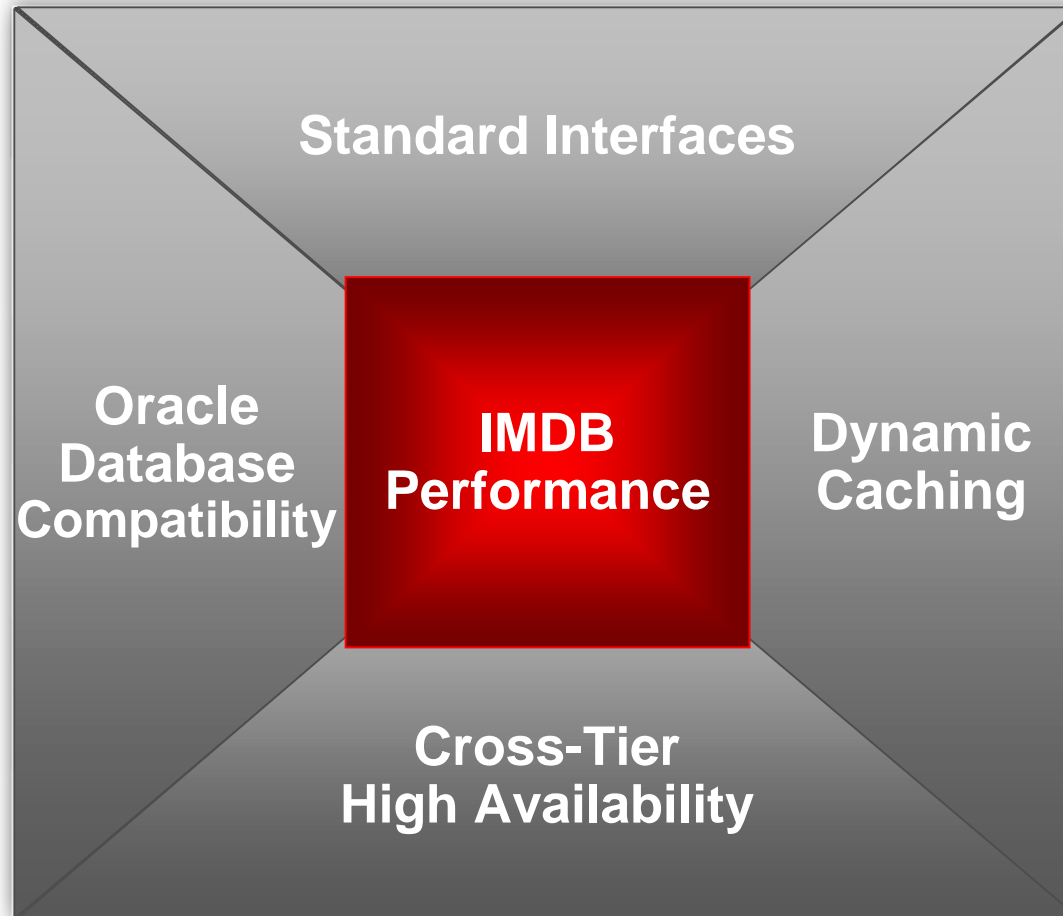


Benefits of In-Memory Database Cache

- Reduced application response time
 - Read and Write transactions on cached tables are completed more quickly
- Very high throughput
 - Low latency yields high throughput
- Reduced workload on the Oracle Database
 - More computing resources available for other workloads
- Run on commodity hardware
 - Available on 20 popular platforms



Offers Unique Capabilities





For More Information

<http://search.oracle.com>

In memory database cache



or

oracle.com/database

ORACLE®